

```
In [ ]: %matplotlib inline
```

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

我確定好了幾個測試主題：

1. 自動控制系統觀念
2. 材料力學概念
3. 程式生成(C#,Python,Matlab)
4. 程式除錯(C#,Python,Matlab)

每個關卡都有分**基礎題**、**進階題**、**應用題**三大部分

以下三種LLM為本次測驗的參賽者：

1. ChatGPT (GPT-4o mini)
2. Claude (Claude 3.7 Sonnet)
3. Gemini (Google Gemini 2.0 Flash)

以下四點為評估模型表現的重點

- 準確性：模型的回答是否正確？是否有錯誤的知識？
- 完整度：回答是否詳細，是否能涵蓋關鍵概念？
- 表達能力：模型的回答是否清晰易懂？
- 創意性：模型是否能提供有趣的觀點或應用？

為確保測試公平性，一律使用新的對話測試

並一律使用以下格式貼 Prompt(為避免報告太長，回答一律盡量簡單)：

請用繁體中文簡短回答以下問題

「(在此貼上問題)」

=====測試開始=====

測試關卡(一)：自動控制系統觀念

基礎題：

- 「請解釋開迴路與閉迴路控制的區別，並舉例說明。」
- 「什麼是狀態空間表示法？它與傳遞函數有何不同？」

進階題：

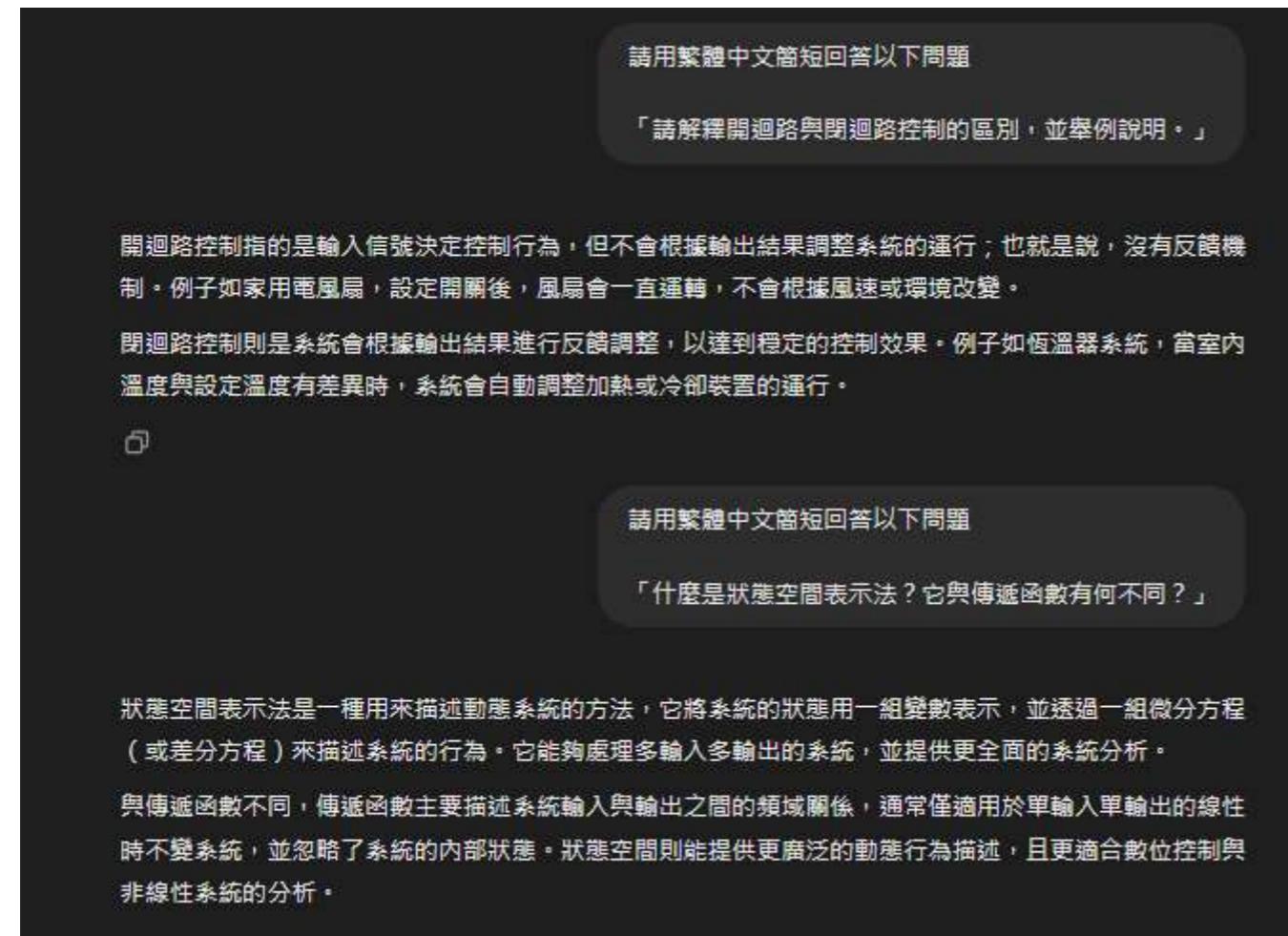
- 「請推導 PID 控制器的數學模型，並說明如何調整參數以改善超調量。」

應用題：

- 「如果要設計一個自動調整車速的控制系統，應該如何選擇適當的控制策略？」

第一位參賽者：ChatGPT

基礎題回答內容截圖，以及輸出內容截圖



進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請推導 PID 控制器的數學模型，並說明如何調整參數以改善超調量。」

PID 控制器的數學模型為：

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{d}{dt} e(t)$$

其中：

- $u(t)$ ：控制輸出
- $e(t)$ ：誤差（期望值與實際值的差）
- K_p ：比例增益
- K_i ：積分增益
- K_d ：微分增益

調整參數以改善超調量的方法：

1. 增大 K_p ：能加速系統響應，但過大可能導致過多的超調。
2. 增大 K_i ：能減少穩態誤差，但過大可能導致系統震盪。
3. 增大 K_d ：能減少超調，穩定系統響應，但過大會使系統過於敏感。

通常，改善超調量的關鍵是適當調整 K_p 和 K_d ，增加 K_d 會有助於減少超調並提高系統穩定性。

應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「如果要設計一個自動調整車速的控制系統，應該如何選擇適當的控制策略？」

設計自動調整車速的控制系統時，可以考慮以下控制策略：

1. **PID 控制**：PID 控制器能根據車速誤差進行調整，通過適當調整比例、積分和微分增益來穩定車速並減少誤差，適合簡單且穩定的車速控制。
2. **模糊控制**：如果車速變化較為複雜，且環境因素（如坡度、交通狀況）會影響車速，模糊控制能根據模糊規則進行調整，適應性強，能處理不確定性和非線性問題。
3. **自適應控制**：在車速變化範圍廣泛或環境條件不穩定時，自適應控制可以根據車輛狀況自動調整控制參數，保持穩定性和精度。
4. **模型預測控制 (MPC)**：根據車輛動態模型預測車速並進行優化調整，適合於複雜的車速控制問題，能處理路況、車輛負載等約束。

選擇控制策略時，應根據車速變化的特性、環境因素和控制精度要求來決定最合適的方案。一般來說，PID 控制較為簡單，適合大多數情況；但若環境複雜，則模糊控制或自適應控制會更有效。



總評價：

- 內容準確性：5/5
- 內容完整度：4/5
- 表達能力：5/5
- 創意性：4/5

第二位參賽者：Claude

基礎題回答內容截圖，以及輸出內容截圖

Differences Between Open-Loop and Closed-Loop Control ▾

嚴 請用繁體中文簡短回答以下問題

「請解釋開迴路與閉迴路控制的區別，並舉例說明。」

Edit

開迴路控制與閉迴路控制的主要區別在於反饋機制的有無。

開迴路控制系統沒有反饋機制，系統輸出不會影響控制過程。例如：電飯煲的定時開關、紅綠燈的定時控制系統。這類系統結構簡單，成本低，但無法對外部干擾作出調整。

閉迴路控制系統具有反饋機制，會不斷監測實際輸出並與預期結果比較，然後根據偏差調整輸入。例如：空調系統（溫度感測器監測室內溫度並調整運行狀態）、汽車的巡航控制系統。這類系統能夠自動修正誤差，提高精確度和穩定性，但結構較複雜且成本較高。

嚴 請用繁體中文簡短回答以下問題

「什麼是狀態空間表示法？它與傳遞函數有何不同？」

狀態空間表示法是一種描述動態系統的數學模型，使用一組一階微分方程式來描述系統的輸入、輸出及內部狀態變數之間的關係。

與傳遞函數的主要區別：

1. 表達方式：狀態空間使用矩陣形式描述系統，而傳遞函數使用拉普拉斯變換的比值形式。
2. 適用範圍：狀態空間適合描述多輸入多輸出(MIMO)系統，傳遞函數主要適用於單輸入單輸出(SISO)系統。
3. 時域與頻域：狀態空間在時域分析系統，傳遞函數則在頻域分析系統。
4. 內部特性：狀態空間能夠表達系統內部狀態變數及其動態特性，傳遞函數僅描述輸入與輸出之間的關係，不能直接反映系統內部結構。
5. 初始條件：狀態空間可以考慮系統的初始條件，而傳遞函數通常假設零初始條件。

嚴 請用繁體中文簡短回答以下問題

Reply to Claude...



Claude 3.7 Sonnet ▾ Choose style ▾

進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題
「請推導 PID 控制器的數學模型，並說明如何調整參數以改善超調量。」

PID控制器的數學模型如下：

對於誤差信號 $e(t)$ ，PID控制器的輸出 $u(t)$ 為：

$$u(t) = K_p \cdot e(t) + K_i \cdot \int e(t) dt + K_d \cdot \frac{de(t)}{dt}$$

其中：

- K_p 是比例增益係數
- K_i 是積分增益係數
- K_d 是微分增益係數

參數調整以改善超調量：

1. 減小 K_p ：過大的比例增益會導致系統響應過快而產生超調。適當降低 K_p 可以減少超調量，但會使系統響應變慢。
2. 增加 K_d ：微分項對誤差變化率作出反應，能夠提前預測系統未來趨勢並施加阻尼作用。適當增加 K_d 可以抑制超調，但過大的 K_d 會放大噪聲並使系統不穩定。
3. 調整 K_i ：積分項主要用於消除穩態誤差，但過大的 K_i 會增加超調量。在減少超調的情況下，可以適當降低 K_i 。
4. 設定合理的 K_p/K_i 比值：這個比值對系統動態特性有重要影響，適當調整可以在維持響應速度的同時減少超調。

調整策略通常是先設定 K_p 獲得基本響應特性，再加入 K_d 減少超調，最後引入 K_i 消除穩態誤差。

應用題回答內容截圖，以及輸出內容截圖

嚴

請用繁體中文簡短回答以下問題

「如果要設計一個自動調整車速的控制系統，應該如何選擇適當的控制策略？」

Edit

設計自動調整車速的控制系統時，可考慮以下控制策略選擇方案：

1. PID控制：適合基本的巡航控制系統，可提供穩定的速度控制。比例項(P)處理即時誤差，積分項(I)消除穩態誤差，微分項(D)預測並減少超調。
2. 自適應控制：能根據路況、負載變化等自動調整控制參數，適應不同駕駛環境。
3. 模糊控制：利用模糊邏輯處理複雜的駕駛條件，可模擬人類駕駛員的決策過程，處理非線性和不確定性。
4. 預測控制(MPC)：預測未來車輛行為並優化控制輸出，適合處理前方路況變化、限速區域等複雜情境。
5. 智能控制：結合機器學習，可隨時間學習優化控制策略，適應用戶駕駛習慣。

選擇因素包括：

- 系統複雜度與成本
- 安全性要求
- 控制精度需求
- 環境適應性
- 人機交互需求

實際應用中，通常採用混合策略，如PID作為基礎，結合自適應或模糊控制增強系統性能，並加入適當冗餘設計確保安全性。

Retry



Claude can make mistakes. Please double-check responses.

Reply to Claude...



Claude 3.7 Sonnet ▾ Choose style ▾

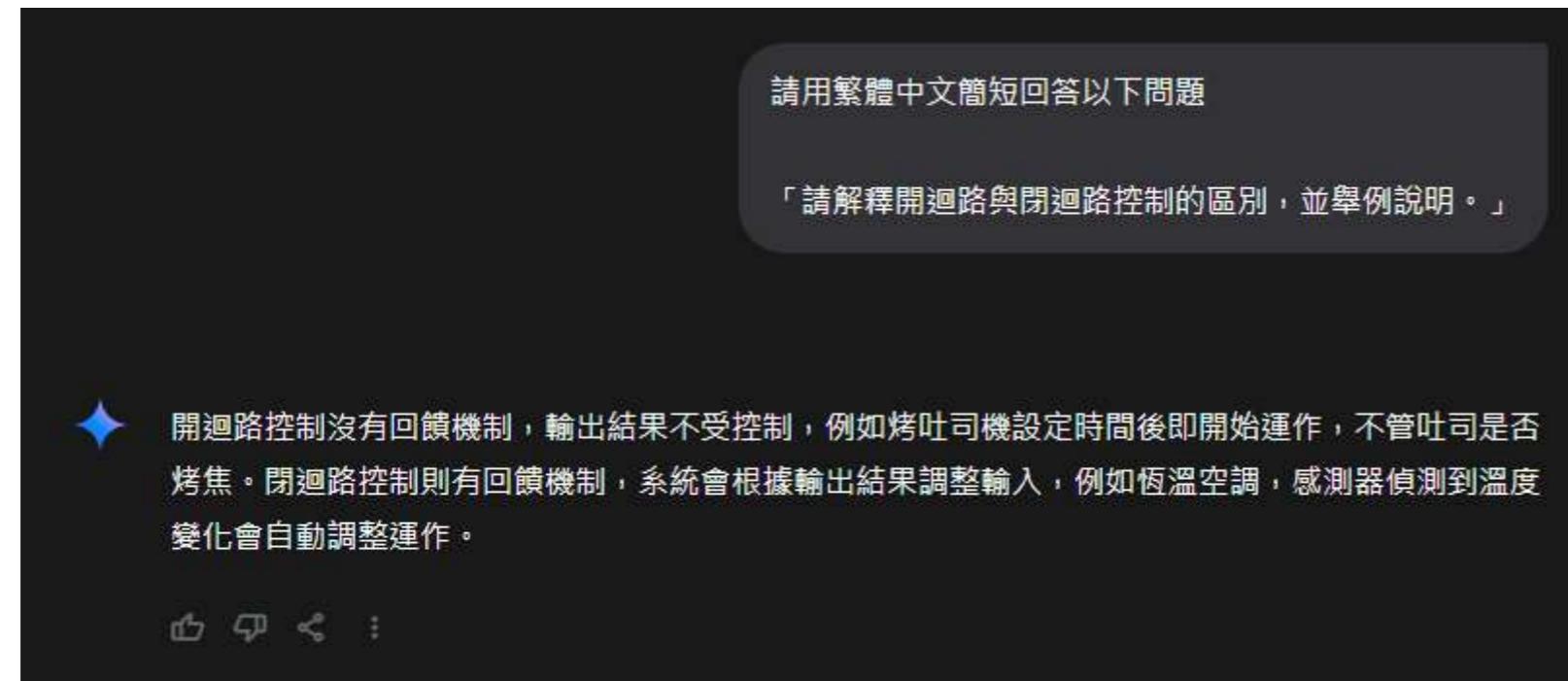
總評價：

- 內容準確性：5/5
- 內容完整度：5/5

- 表達能力 : 5/5
- 創意性 : 5/5

第三位參賽者 : Gemini

基礎題回答內容截圖，以及輸出內容截圖



請用繁體中文簡短回答以下問題

「什麼是狀態空間表示法？它與傳遞函數有何不同？」

- ◆ 狀態空間表示法是一種用來描述系統動態行為的數學模型。它使用一組一階微分方程式來表示系統的狀態、輸入和輸出之間的關係。

狀態空間表示法與傳遞函數的主要區別在於：

- 表示方式：
 - 狀態空間表示法使用矩陣和向量來描述系統。
 - 傳遞函數則使用拉普拉斯變換來描述系統的輸入輸出關係。
- 適用範圍：
 - 狀態空間表示法適用於多輸入多輸出 (MIMO) 系統和非線性系統。
 - 傳遞函數主要適用於單輸入單輸出 (SISO) 的線性時不變系統。
- 資訊量：
 - 狀態空間表示法能提供系統內部的詳細狀態資訊。
 - 傳遞函數僅描述輸入輸出之間的關係，不包含內部狀態資訊。

簡單來說，狀態空間表示法更全面，傳遞函數更簡潔。

進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請推導 PID 控制器的數學模型，並說明如何調整參數以改善超調量。」

◆ PID 控制器的數學模型為：

$$u(t) = K_p \cdot e(t) + K_i \cdot \int e(\tau) d\tau + K_d \cdot \frac{de(t)}{dt}$$

其中：

- $u(t)$ 是控制輸出
- $e(t)$ 是誤差（設定值與實際值之差）
- K_p 是比例增益
- K_i 是積分增益
- K_d 是微分增益

調整參數以改善超調量：

- 降低 K_p ：減少比例作用，降低系統響應速度，從而減少超調量。
- 降低 K_i ：減少積分作用，降低穩態誤差，但可能增加超調量。
- 增加 K_d ：增加微分作用，預測誤差變化率，抑制超調量。

調整參數時，通常需要根據實際系統響應進行多次調整，以達到最佳效果。

應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「如果要設計一個自動調整車速的控制系統，應該如何選擇適當的控制策略？」

◆ 設計自動調整車速的控制系統時，應考慮以下控制策略：

- PID 控制：
 - 適用於大多數車速控制場景，能有效平衡響應速度和穩定性。
 - 透過調整比例、積分和微分參數，可實現精確的車速控制。
- 模型預測控制 (MPC)：
 - 適用於複雜路況和需要預測未來車速的場景，例如自動駕駛。
 - 能夠根據車輛模型和環境資訊，預測未來車速變化，並進行最佳化控制。
- 模糊控制：
 - 適用於難以建立精確數學模型的場景，例如在不同路況下調整車速。
 - 透過模糊邏輯，能夠模擬人類駕駛員的經驗，實現平滑的車速控制。

選擇控制策略時，應考慮以下因素：

- 系統的複雜性
- 控制精度要求
- 系統的響應速度
- 系統的穩定性

總評價：

- 內容準確性：5/5
- 內容完整度：3/5
- 表達能力：4/5
- 創意性：3/5

測試關卡(二)：材料力學概念

基礎題：

- 「請解釋應力 (Stress) 與應變 (Strain) 的概念，並提供示例。」
- 「什麼是胡克定律？請寫出數學表達式並說明變數的意義。」

進階題：

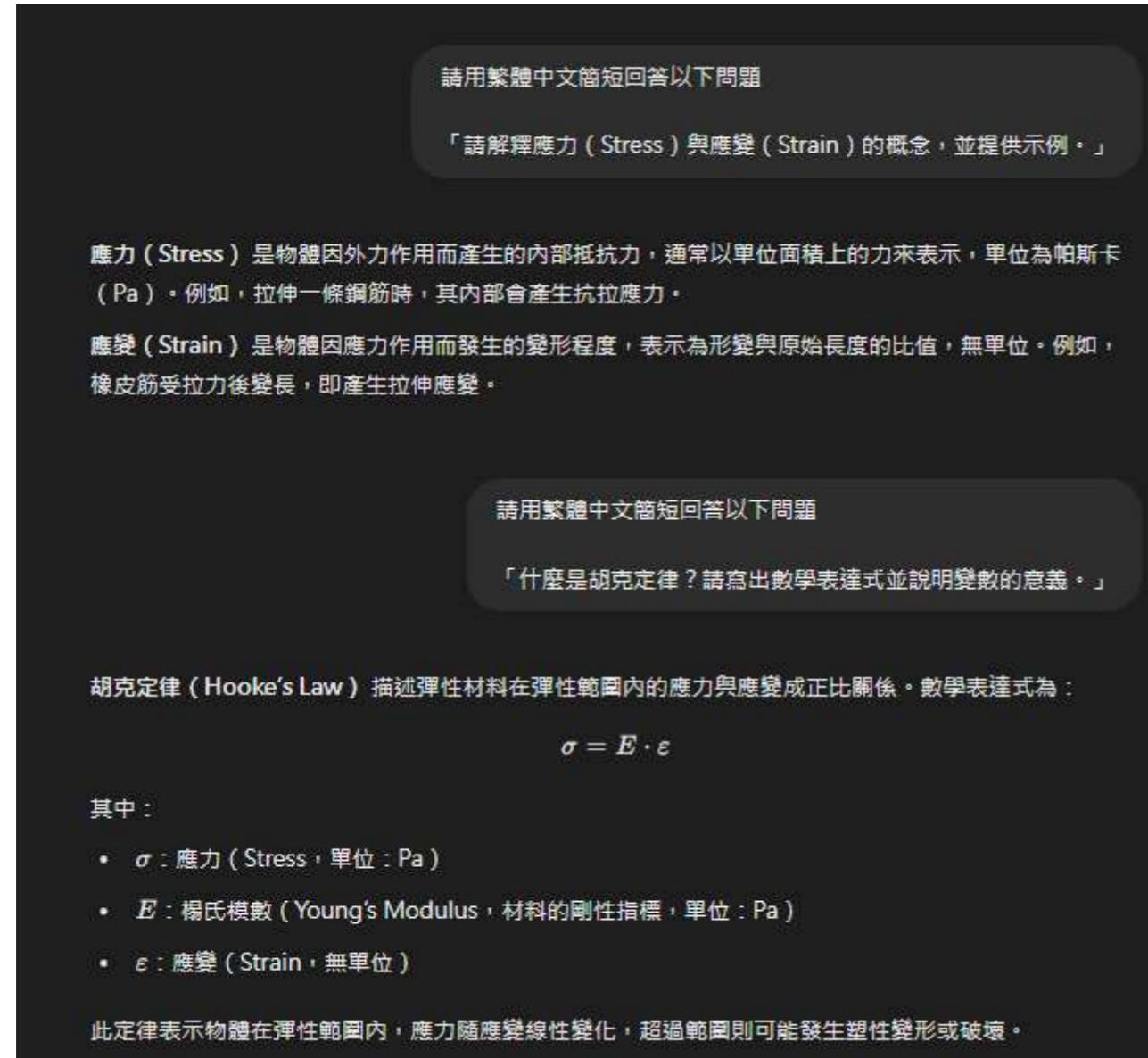
- 「請解釋剪應力與彎矩之間的關係，並舉例說明其應用。」

應用題：

- 「設計一個支撐結構，使其在最大負載下的變形量最小，你會如何選擇材料和形狀？」

第一位參賽者：ChatGPT

基礎題回答內容截圖，以及輸出內容截圖



進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請解釋剪應力與彎矩之間的關係，並舉例說明其應用。」

剪應力與彎矩的關係

剪應力 (Shear Stress) 與彎矩 (Bending Moment) 在樑的受力分析中相互關聯，其基本關係為：

$$\frac{dM}{dx} = V$$

其中：

- M 為彎矩 (Bending Moment)
- V 為剪力 (Shear Force)
- x 為樑的長度坐標

此關係表示彎矩的變化率等於剪力，若剪力為零，則彎矩為極值點。

應用舉例

1. 橋樑工程：在橋樑設計中，需考慮剪力與彎矩分佈，以確保橋樑承受車輛荷重時不會失效。
2. 機械結構：如起重機吊臂，受力點附近的剪應力與彎矩影響結構強度。
3. 建築樑柱：建築中的樓板樑需計算剪力與彎矩，以選擇適當的材料與截面形狀，提高承載能力。

應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「設計一個支撐結構，使其在最大負載下的變形量最小，你會如何選擇材料和形狀？」

設計支撐結構以最小化變形量的策略

1. 選擇高剛性材料

- 選擇高楊氏模數 (E) 的材料，如鋼 (Steel)、鈦合金 (Titanium Alloy) 或碳纖維複合材料 (Carbon Fiber Composite)，以減少彎曲變形。

2. 最佳化截面形狀 (提高慣性矩 I)

- I型樑 (I-Beam)：適用於承受彎矩的結構，如橋樑與建築樑。
- 箱型樑 (Box Beam)：高抗扭剛性，適用於機械框架與飛機結構。
- 圓管 (Hollow Circular Tube)：提供高強度與抗彎能力，常用於塔架與支撐柱。

3. 減少自由端懸臂，優化邊界條件

- 固定支撐比簡支支撐變形小，若可行，應選用固定端支撐方式。

4. 減少結構長度 (L) 或增加支撐點

- 變形量與樑長度的三次方或四次方成正比，縮短跨度或增加支撐可大幅減少變形。

5. 增加構造剛性 (例如加強筋或桁架設計)

- 使用**三角桁架 (Truss) 或加強筋 (Rib Reinforcement) **來分散應力，減少局部變形。

總評價：

- 內容準確性：5/5
- 內容完整度：4/5
- 表達能力：4/5
- 創意性：4/5

第二位參賽者：Claude

基礎題回答內容截圖，以及輸出內容截圖

嚴

請用繁體中文簡短回答以下問題

「請解釋應力 (Stress) 與應變 (Strain) 的概念，並提供示例。」

◇ Edit

應力 (Stress) 與應變 (Strain) 是材料力學中的基本概念：

應力是指材料內部對外加力的反應，表示單位面積上所承受的力。計量單位通常為帕斯卡 (Pa) 或牛頓/平方米 (N/m²)。例如，當我們懸掛一個重物在鋼絲上時，鋼絲內部會產生應力來平衡外部的拉力。

應變則表示材料在受力後的形變程度，是材料變形量與原始尺寸的比值，是一個無單位的量。

例如，橡皮筋在拉伸時會變長，其長度的相對變化就是應變。

日常生活示例：

- 彈簧被壓縮時，其內部產生應力，而其長度縮短的比例則是應變
- 建築結構中的樑柱承受重量時產生應力，微小的形變則是應變

應力與應變之間的關係被描述為材料的彈性模量或楊氏模量，這反映了材料的剛性特性。

嚴

請用繁體中文簡短回答以下問題

「什麼是胡克定律？請寫出數學表達式並說明變數的意義。」

◇ Edit

胡克定律 (Hooke's Law) 描述了彈性材料在彈性限度內，應力與應變成正比的關係。

數學表達式：

$$\sigma = E \cdot \epsilon$$

其中：

- σ (sigma) 代表應力，單位為帕斯卡 (Pa)
- ϵ (epsilon) 代表應變，無單位
- E 代表楊氏模量 (或彈性模量)，單位為帕斯卡 (Pa)

這個定律說明材料在彈性形變範圍內，施加的應力越大，產生的應變也越大，且兩者成線性比例關係。楊氏模量 E 是材料的固有特性，反映了材料抵抗彈性變形的能力— E 值越大，表示材料越硬，相同應力下產生的形變越小。

胡克定律只適用於材料的彈性區間，超過彈性限度後，應力與應變的關係將不再遵循這個線性關係。

進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題
「請解釋剪應力與彎矩之間的關係，並舉例說明其應用。」

剪應力與彎矩之間的關係是結構力學中的重要概念：

剪應力 (V) 與彎矩 (M) 的微分關係： $dM/dx = V$ ，即彎矩對位置的導數等於剪應力；或者說，剪應力是彎矩沿梁長方向的變化率。

積分關係： $M(x) = \int V(x)dx + C$ ，即彎矩是剪應力沿梁長積分的結果。

應用實例：

- 橋樑設計：工程師必須計算橋樑各點的剪應力和彎矩分佈，以確保結構安全。橋樑中部通常彎矩最大而剪應力較小，支撐點附近則剪應力較大。
- 高樓建築：在設計樓板和橫梁時，需要分析彎矩和剪應力分佈，以選擇適當的材料和尺寸。
- 飛機機翼：機翼受到升力產生彎矩和剪應力，設計師必須計算這些力來確保機翼強度足夠但重量不過大。

理解這種關係有助於結構設計者預測可能的失效位置，並適當地加強結構的關鍵部位。

應用題回答內容截圖，以及輸出內容截圖

嚴

請用繁體中文簡短回答以下問題

「設計一個支撐結構，使其在最大負載下的變形量最小，你會如何選擇材料和形狀？」

設計低變形高負載支撐結構時，材料和形狀選擇關鍵考量如下：

材料選擇：

- 高楊氏模量材料：優先考慮鋼、鈦合金或特種碳纖維複合材料，因其具有優異的強度/重量比
- 材料剛性：選擇高剛性材料以降低應變量
- 抗屈服強度：確保材料在設計負載下不會永久變形
- 疲勞限度：若結構需承受循環載荷，必須考慮材料的疲勞特性

形狀設計：

- 截面慣性矩最大化：採用I型梁、箱型結構或管狀截面，增加結構對彎曲的抵抗能力
- 三角形桁架：利用三角穩定性原理分散應力
- 材料分佈：將材料集中於承受最大應力的區域
- 連續性支撐：避免應力集中點
- 使用空心結構：減輕重量同時保持結構剛性

實務應用中，常結合有限元分析(FEA)進行優化設計，在強度、重量、成本和製造難度間取得平衡。特殊情況下，可考慮使用複合材料或三明治結構以獲得最佳的剛性/重量比。

總評價：

- 內容準確性：5/5
- 內容完整度：5/5
- 表達能力：4/5
- 創意性：5/5

第三位參賽者：Gemini

基礎題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請解釋應力 (Stress) 與應變 (Strain) 的概念，並提供示例。」

◆

- 應力 (Stress)：
 - 指物體單位面積上所承受的力，也就是物體內部的受力強度。
 - 例如：拉扯橡皮筋時，橡皮筋內部產生的抵抗拉力的力。
- 應變 (Strain)：
 - 指物體在外力作用下產生的變形程度，通常表示為變形量與原始尺寸的比值。
 - 例如：拉扯橡皮筋時，橡皮筋長度增加的比例。

請用繁體中文簡短回答以下問題

「什麼是胡克定律？請寫出數學表達式並說明變數的意義。」

◆

- 胡克定律是描述固體材料受力後，應力與應變之間線性關係的定律。也就是說，在彈性限度內，物體受到的外力與其產生的形變成正比。
- 數學表達式：
 - $F = -kx$
 - 其中：
 - F：彈簧產生的力
 - k：彈簧的勁度係數，表示彈簧的硬度
 - x：彈簧的變形量

進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請解釋剪應力與彎矩之間的關係，並舉例說明其應用。」

- ◆ • 剪應力與彎矩的關係：
 - 剪應力與彎矩在樑或桿件等結構中密切相關。當一個物體受到橫向載荷時，會同時產生剪應力和彎矩。
 - 剪應力是指物體內部平行於截面的力，而彎矩是指使物體彎曲的力矩。
 - 簡言之，剪力會導致剪應力的產生，而彎矩則會導致彎曲應力的產生，這兩者共同作用於結構中。
- 應用舉例：
 - 橋樑設計：
 - 在設計橋樑時，工程師必須計算橋樑承受的剪應力和彎矩，以確保橋樑在車輛和其他載荷的作用下不會發生斷裂或變形。
 - 建築結構：
 - 建築物中的樑和柱也受到剪應力和彎矩的作用。設計時需要考慮這些應力，以確保建築物的安全性和穩定性。
 - 機械零件：
 - 例如：在設計機械零件（如軸）時，需要分析其承受的剪應力和彎矩，以防止零件在運作過程中發生損壞。

應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「設計一個支撐結構，使其在最大負載下的變形量最小，
你會如何選擇材料和形狀？」

◆ 要設計一個在最大負載下變形量最小的支撐結構，材料和形狀的選擇至關重要。以下是一些考量因素：

材料選擇：

- 高強度材料：
 - 選擇具有高楊氏模量和高屈服強度的材料，例如鋼、鈦合金或碳纖維複合材料。這些材料在承受相同負載時，變形量較小。
- 材料特性：
 - 考慮材料的密度、耐腐蝕性和成本等因素。例如，鋼材強度高、成本相對較低，但容易生鏽；碳纖維複合材料強度極高，但成本較高。

形狀選擇：

- 結構幾何：
 - 採用能夠有效分散應力的結構形狀，例如桁架、拱形或殼體結構。這些結構能夠將負載均勻分佈，減少局部應力集中。
- 截面形狀：
 - 選擇具有較大慣性矩的截面形狀，例如工字梁或箱形梁。這些形狀能夠提高結構的抗彎能力，減少彎曲變形。
- 優化設計：
 - 利用電腦輔助設計 (CAD) 和有限元素分析 (FEA) 軟體，對結構形狀進行優化，以達到最小變形量的目標。

問問 Gemini



Deep Research



- 優化設計：
 - 利用電腦輔助設計 (CAD) 和有限元素分析 (FEA) 軟體，對結構形狀進行優化，以達到最小變形量的目標。

具體應用舉例：

- 橋樑：
 - 採用鋼結構或預應力混凝土結構，並使用桁架或拱形結構來分散負載。
- 建築物：
 - 使用鋼骨或鋼筋混凝土結構，並採用合理的樑柱佈置和截面尺寸。
- 飛機：
 - 使用輕質高強度的碳纖維複合材料，並採用優化的機翼和機身形狀。

總之，材料和形狀的選擇需要綜合考慮負載條件、環境因素和成本等因素，以達到最佳的設計效果。

回 此 C < :

總評價：

- 內容準確性：4/5
- 內容完整度：3/5
- 表達能力：5/5
- 創意性：5/5

測試關卡(三)：程式生成 (C# 、 Python 、 Matlab)

基礎題：

- 「請撰寫一個 Python 程式來計算費氏數列 (Fibonacci) 。」
- 「請用 Matlab 繪製函數 $y = \sin(x)$ 的曲線。」

進階題：

- 「請用 Matlab 設計一個控制系統模擬，並分析其步階響應 (step response) 。」

應用題：

- 「請用 Python 設計一個簡單的 GUI 應用程式，讓使用者輸入數據並計算 BMI。」

第一位參賽者：ChatGPT

基礎題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題
「請撰寫一個 Python 程式來計算費氏數列（Fibonacci）。」

以下是計算費氏數列的 Python 程式：

```
python
```

```
def fibonacci(n):
    a, b = 0, 1
    for _ in range(n):
        a, b = b, a + b
    return a

# 範例使用
n = 10
print(f"第 {n} 項費氏數列為: {fibonacci(n)}")
```

```
In [ ]: def fibonacci(n):
    a, b = 0, 1
    for _ in range(n):
        a, b = b, a + b
    return a

# 範例使用
n = 10
print(f"第 {n} 項費氏數列為: {fibonacci(n)}")
```

第 10 項費氏數列為: 55

請用繁體中文簡短回答以下問題

「請用 Matlab 繪製函數 $y = \sin(x)$ 的曲線。」

在 Matlab 中，可以使用以下程式碼繪製 $y = \sin(x)$ 的曲線：

A screenshot of a MATLAB code editor window. The title bar says "matlab". The code area contains the following MATLAB script:

```
x = linspace(-pi, pi, 100);
y = sin(x);
plot(x, y);
xlabel('x');
ylabel('y');
title('y = sin(x)');
grid on;
```

The window has standard OS X-style controls at the bottom.

```
untitled.m %chatGPT
1 x = linspace(-pi, pi, 100);
2 y = sin(x);
3 plot(x, y);
4 xlabel('x');
5 ylabel('y');
6 title('y = sin(x)');
7 grid on;
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
```

Figure 1

File Edit View Insert Tools Desktop Window Help

New to MATLAB

fx >>

The figure shows a sine wave plotted against x. The x-axis is labeled 'x' and ranges from -4 to 4. The y-axis is labeled 'y' and ranges from -1 to 1. The plot title is 'y = sin(x)'. The curve starts at (0,0), goes down to a minimum of approximately -0.8 at x ≈ -1.57, crosses the x-axis at x ≈ -3.14, reaches a maximum of approximately 0.8 at x ≈ 1.57, and returns to the x-axis at x ≈ 3.14.

進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

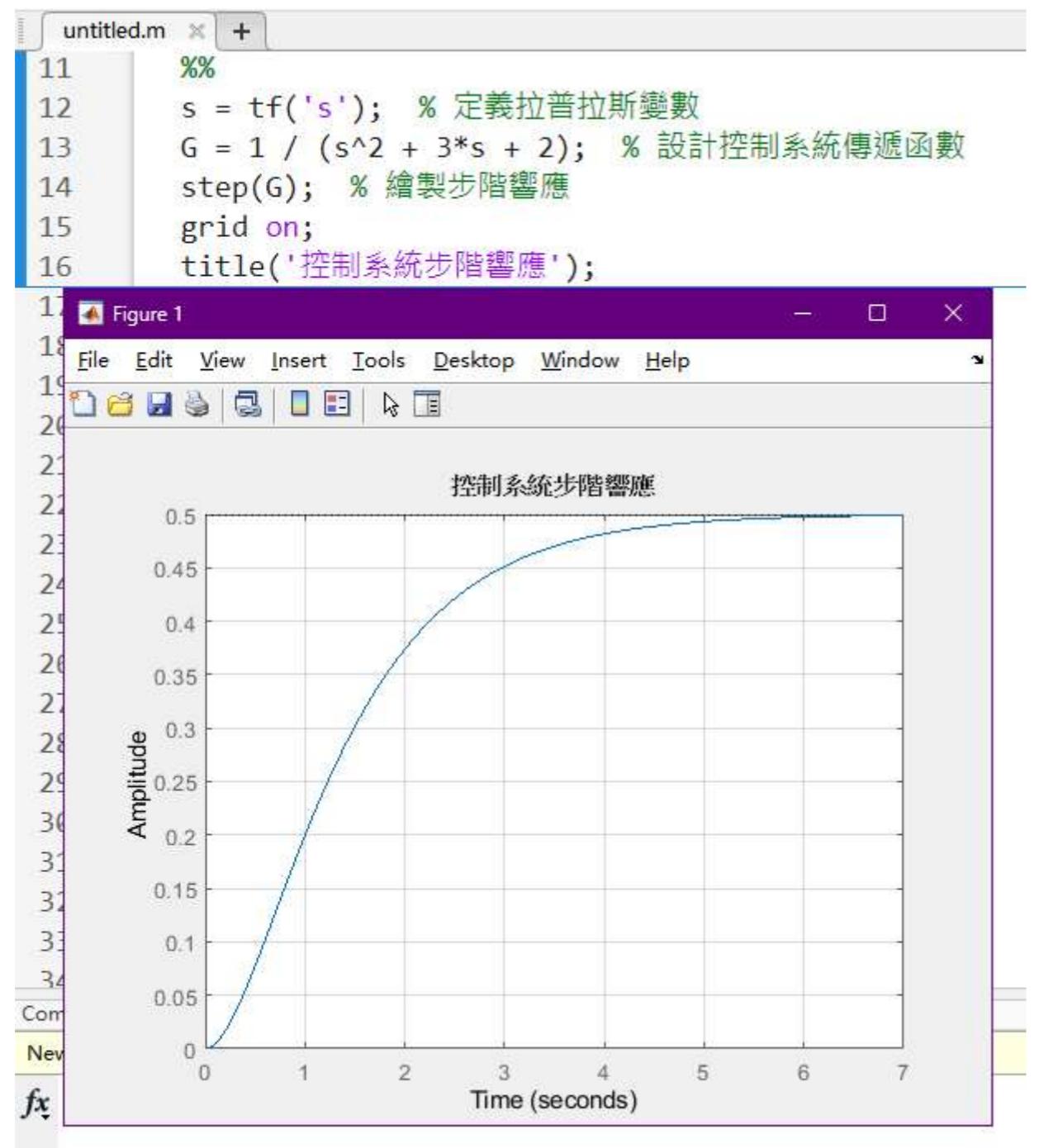
「請用 Matlab 設計一個控制系統模擬，並分析其步階響應（step response）。」

在 MATLAB 中，可使用以下程式碼設計一個簡單的控制系統並分析其步階響應：

```
matlab:          ⚠️ 複製    ⚡ 執行  
s = tf('s'); % 定義拉普拉斯變數  
G = 1 / (s^2 + 3*s + 2); % 設計控制系統傳遞函數  
step(G); % 繪製步階響應  
grid on;  
title('控制系統步階響應');
```

此程式碼建立了一個二階系統並繪製其步階響應曲線。





應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請用 Python 設計一個簡單的 GUI 應用程式，讓使用者輸入數據並計算 BMI。」

以下是用 Python 和 `tkinter` 設計的簡單 GUI 應用程式，讓使用者輸入身高和體重並計算 BMI：

```
python
import tkinter as tk

def calculate_bmi():
    weight = float(entry_weight.get())
    height = float(entry_height.get()) / 100 # 將身高轉換為公尺
    bmi = weight / (height ** 2)
    result_label.config(text=f"BMI: {bmi:.2f}")

# 建立主視窗
window = tk.Tk()
window.title("BMI 計算器")

# 身高輸入
label_height = tk.Label(window, text="身高 (公分):")
label_height.pack()

entry_height = tk.Entry(window)
entry_height.pack()

# 體重輸入
label_weight = tk.Label(window, text="體重 (公斤):")
label_weight.pack()

entry_weight = tk.Entry(window)
entry_weight.pack()
```

```
# 體重輸入
label_weight = tk.Label(window, text="體重 (公斤):")
label_weight.pack()

entry_weight = tk.Entry(window)
entry_weight.pack()

# 計算按鈕
calculate_button = tk.Button(window, text="計算 BMI", command=calculate_bmi)
calculate_button.pack()

# 顯示結果
result_label = tk.Label(window, text="BMI: ")
result_label.pack()

# 啟動主視窗
window.mainloop()
```

這段程式碼建立了一個 GUI 應用程式，讓使用者輸入身高和體重，並計算並顯示 BMI。



The screenshot shows a code editor with a Python script named ChatG.py. The script uses the Tkinter library to create a BMI calculator application. The application window has a title 'BMI 計算器' and contains two entry fields for height (177.5) and weight (61.8), a button labeled '計算 BMI', and a result label showing BMI: 19.62.

```
1 import tkinter as tk
2
3     Codeium: Refactor | Explain | Generate Docstring | X
4 def calculate_bmi():
5     weight = float(entry_weight.get())
6     height = float(entry_height.get()) / 100 # 將身高轉換為公尺
7     bmi = weight / (height ** 2)
8     result_label.config(text=f"BMI: {bmi:.2f}")
9
10    # 建立主視窗
11    window = tk.Tk()
12    window.title("BMI 計算器")
13
14    # 身高輸入
15    label_height = tk.Label(window, text="身高 (公分):")
16    label_height.pack()
17
18    entry_height = tk.Entry(window)
19    entry_height.pack()
20
21    # 體重輸入
22    label_weight = tk.Label(window, text="體重 (公斤):")
23    label_weight.pack()
24
25    entry_weight = tk.Entry(window)
26    entry_weight.pack()
27
28    # 計算按鈕
29    calculate_button = tk.Button(window, text="計算 BMI", command=calculate_bmi)
30    calculate_button.pack()
31
32    # 顯示結果
33    result_label = tk.Label(window, text="BMI: ")
34    result_label.pack()
35
36    # 啟動主視窗
37    window.mainloop()
```

總評價：

- 內容準確性：4/5
- 內容完整度：3/5
- 表達能力：5/5
- 創意性：4/5

第二位參賽者：Claude

基礎題回答內容截圖，以及輸出內容截圖

嚴 請用繁體中文簡短回答以下問題
「請撰寫一個 Python 程式來計算費氏數列（Fibonacci）。」

以下是一個計算費氏數列的 Python 程式：

```
python
def fibonacci(n):
    if n <= 0:
        return "請輸入正整數"
    elif n == 1:
        return 0
    elif n == 2:
        return 1
    else:
        a, b = 0, 1
        for _ in range(3, n+1):
            a, b = b, a + b
        return b

# 測試程式
n = 10
print(f"費氏數列第 {n} 項為: {fibonacci(n)}")
```

此程式使用迭代方法計算費氏數列，避免了遞迴可能導致的堆疊溢出問題。您可以修改變數 `n` 的值來計算不同項的費氏數。

ⓘ Claude does not have the ability to run the code it generates yet.

Copy Retry

In []: `def fibonacci(n):
 if n <= 0:
 return "請輸入正整數"
 elif n == 1:
 return 0
 elif n == 2:
 return 1
 else:
 a, b = 0, 1
 for _ in range(3, n+1):
 a, b = b, a + b`

```
return b

# 測試程式
n = 10
print(f"費氏數列第 {n} 項為: {fibonacci(n)}")
```

費氏數列第 10 項為: 34

請用繁體中文簡短回答以下問題
「請用 Matlab 繪製函數 $y = \sin(x)$ 的曲線。」

以下是用 Matlab 繪製 $y = \sin(x)$ 曲線的程式碼：

```
matlab
% 定義 x 的範圍，從 0 到 2π
x = 0:0.01:2*pi;

% 計算 sin(x)
y = sin(x);

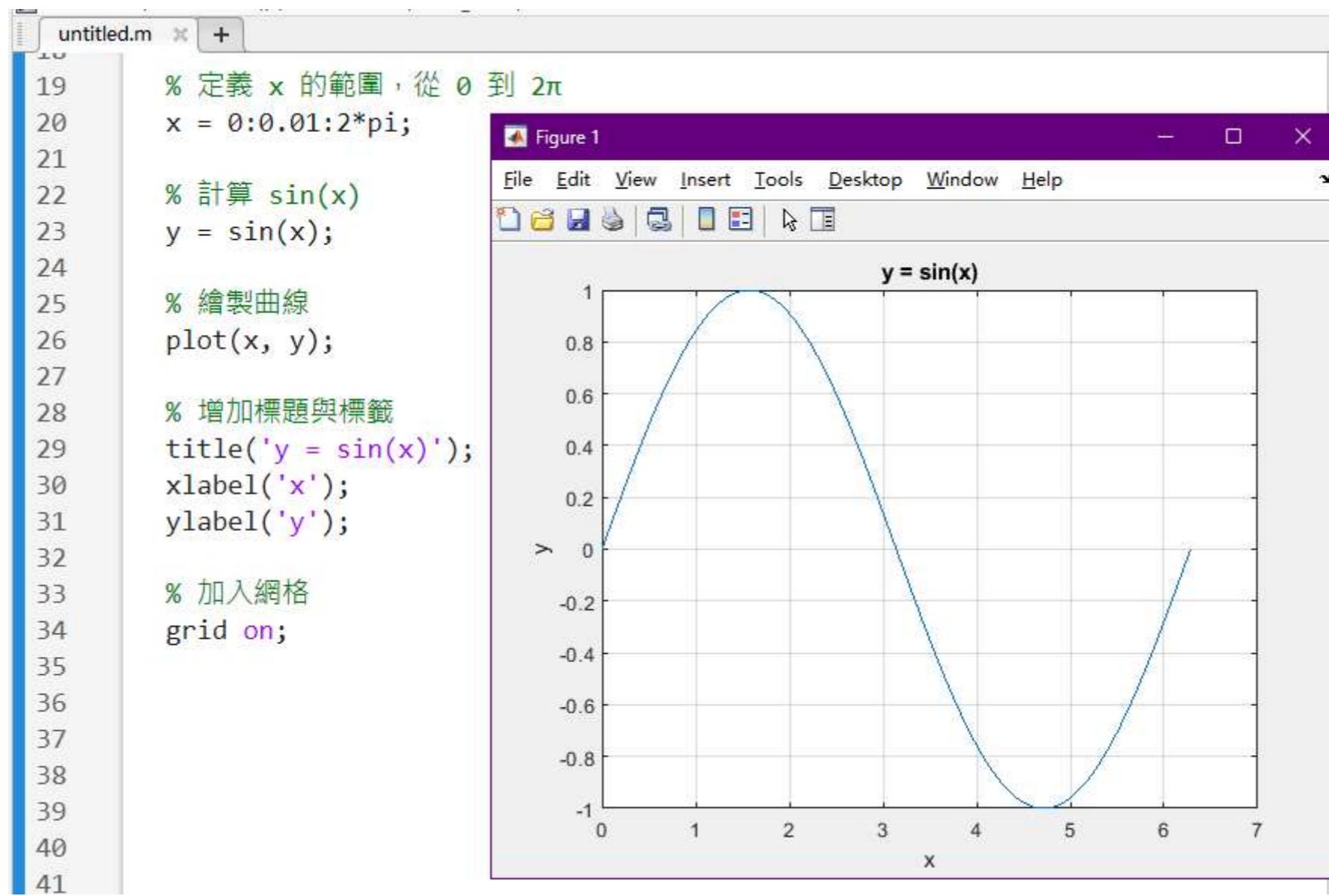
% 繪製曲線
plot(x, y);

% 增加標題與標籤
title('y = sin(x)');
xlabel('x');
ylabel('y');

% 加入網格
grid on;
```

執行此程式碼後，Matlab 會顯示一條完整的正弦曲線，從 0 到 2π 。程式中的 0.01 是步長值，可以調整以改變曲線的平滑度。

ⓘ Claude does not have the ability to run the code it generates yet.



進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請用 Matlab 設計一個控制系統模擬，並分析其步階響應 (step response) 」

以下是用 Matlab 設計控制系統並分析步階響應的程式碼：

```
matlab

% 定義傳遞函數
num = [1]; % 分子項係數
den = [1 2 1]; % 分母項係數 ( $s^2 + 2s + 1$ )
sys = tf(num, den); % 建立傳遞函數  $G(s) = 1/(s^2 + 2s + 1)$ 

% 繪製步階響應
figure;
step(sys);
title('控制系統步階響應');
grid on;

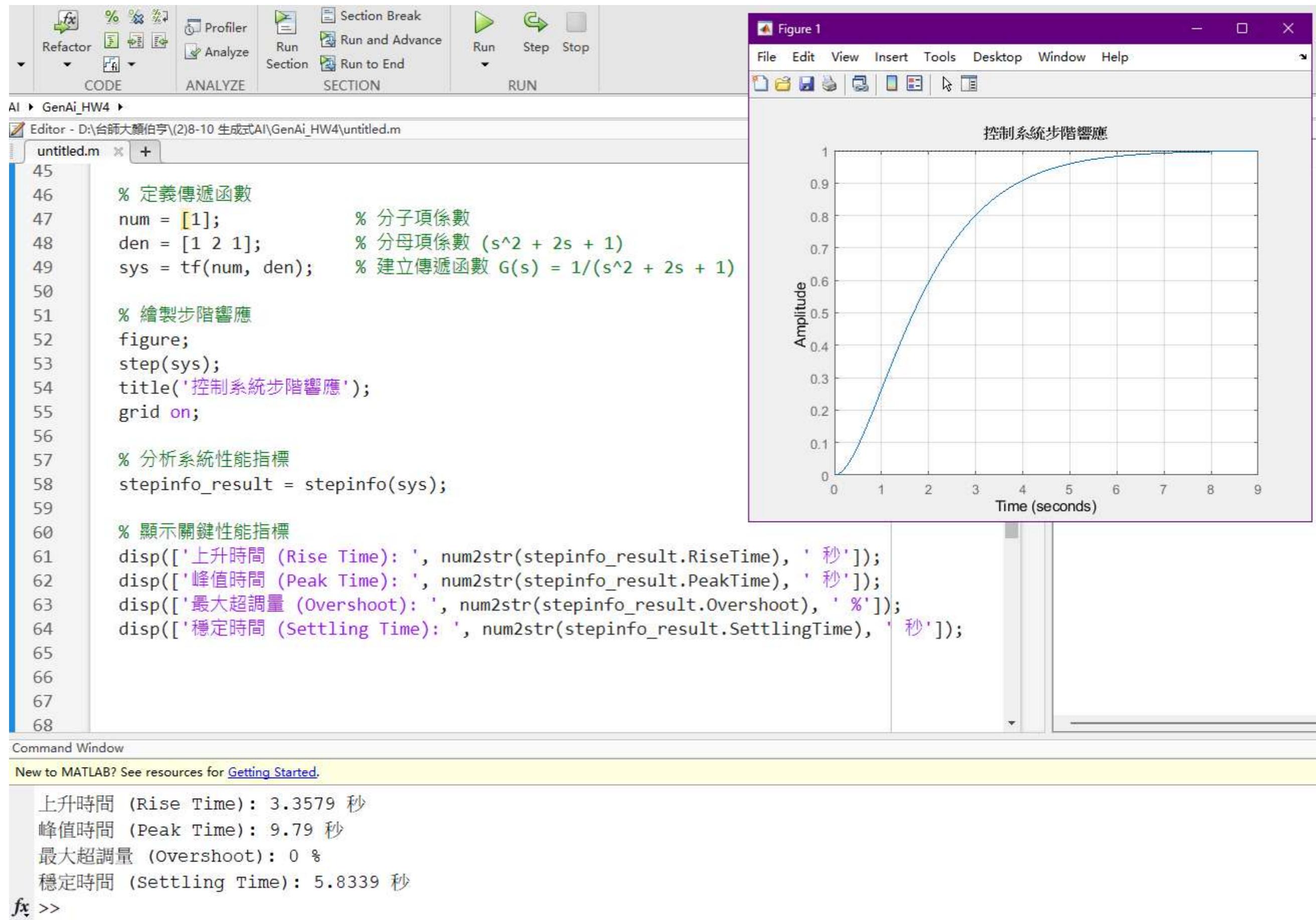
% 分析系統性能指標
stepinfo_result = stepinfo(sys);

% 顯示關鍵性能指標
disp(['上升時間 (Rise Time): ', num2str(stepinfo_result.RiseTime), ' 秒']);
disp(['峰值時間 (Peak Time): ', num2str(stepinfo_result.PeakTime), ' 秒']);
disp(['最大超調量 (Overshoot): ', num2str(stepinfo_result.Overshoot), ' %']);
disp(['穩定時間 (Settling Time): ', num2str(stepinfo_result.SettlingTime), ' 秒']);
```

此程式設計了一個二階系統的傳遞函數 $G(s) = 1/(s^2 + 2s + 1)$ ，並繪製其步階響應曲線。同時，
程式使用 `stepinfo` 函數計算了系統的關鍵性能指標，包括上升時間、峰值時間、最大超調量
和穩定時間。您可以根據實際需求修改傳遞函數的分子和分母係數來模擬不同的控制系統。

① Claude does not have the ability to run the code it generates yet.

Copy Retry



應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題
「請用 Python 設計一個簡單的 GUI 應用程式，讓使用者輸入數據並計算 BMI。」

BMI 計算器 GUI 應用程式
Click to open code

我已經設計了一個簡單的 BMI 計算器 GUI 應用程式，使用 Python 的 tkinter 套件。此應用程式具有以下功能：

1. 提供輸入欄位讓使用者輸入體重（公斤）和身高（公分）
2. 「計算 BMI」按鈕可計算 BMI 值並顯示對應的體重狀態
3. 「清除」按鈕可重置所有輸入和結果
4. 包含錯誤處理功能，確保輸入有效的數字

BMI 計算方式為體重(kg)除以身高(m)的平方，結果會顯示到小數點後一位。體重狀態依照台灣衛福部標準分為：過輕、正常、過重、輕度肥胖、中度肥胖和重度肥胖。

要執行此程式，您需要安裝 Python 和 tkinter 套件（大多數 Python 安裝已包含此套件）。

Claude can make mistakes. Please double-check responses.

Reply to Claude...

Claude 3.7 Sonnet Choose style

← BMI 計算器 GUI 應用程式

```
bmi = weight / (height * height)
self.bmi_result.config(text=f"{bmi:.1f}")

# 判斷體重狀態
if bmi < 18.5:
    status = "過輕"
elif bmi < 24:
    status = "正常"
elif bmi < 27:
    status = "過重"
elif bmi < 30:
    status = "輕度肥胖"
elif bmi < 35:
    status = "中度肥胖"
else:
    status = "重度肥胖"

self.status_result.config(text=status)

except ValueError:
    messagebox.showerror("錯誤", "請輸入有效的數字")

def clear_entries(self):
    self.weight_entry.delete(0, tk.END)
    self.height_entry.delete(0, tk.END)
    self.bmi_result.config(text="")
    self.status_result.config(text="")

if __name__ == "__main__":
    root = tk.Tk()
    app = BMICalculator(root)
    root.mainloop()
```

Last edited just now

Published

```

Claude-bmi-calculator-gui.py
...
Claude-bmi-calculator-gui.py > BMIcalculator > __init__
1 import tkinter as tk
2 from tkinter import messagebox
3
4 class BMIcalculator:
5     def __init__(self, root):
6         self.root = root
7         self.root.title("BMI 計算器")
8         self.root.geometry("300x250")
9         self.root.resizable(False, False)
10
11     # 設定標籤和輸入框
12     tk.Label(root, text="體重 (公斤)").grid(row=0, column=0, padx=10, pady=10, sticky="w")
13     self.weight_entry = tk.Entry(root, width=10)
14     self.weight_entry.grid(row=0, column=1, padx=10, pady=10)
15
16     tk.Label(root, text="身高 (公分)").grid(row=1, column=0, padx=10, pady=10, sticky="w")
17     self.height_entry = tk.Entry(root, width=10)
18     self.height_entry.grid(row=1, column=1, padx=10, pady=10)
19
20     # 計算按鈕
21     calculate_btn = tk.Button(root, text="計算 BMI", command=self.calculate_bmi)
22     calculate_btn.grid(row=2, column=0, columnspan=2, pady=10)
23
24     # 結果顯示區域
25     tk.Label(root, text="BMI 值 :").grid(row=3, column=0, padx=10, pady=5, sticky="w")
26     self.bmi_result = tk.Label(root, text="", width=10)
27     self.bmi_result.grid(row=3, column=1, padx=10, pady=5)
28
29     tk.Label(root, text="體重狀態 :").grid(row=4, column=0, padx=10, pady=5, sticky="w")
30     self.status_result = tk.Label(root, text="", width=10)
31     self.status_result.grid(row=4, column=1, padx=10, pady=5)
32
33     # 滅除按鈕
34     clear_btn = tk.Button(root, text="清除此", command=self.clear_entries)
35     clear_btn.grid(row=5, column=0, columnspan=2, pady=10)
36
37     def calculate_bmi(self):
38         try:
39             weight = float(self.weight_entry.get())
40             height = float(self.height_entry.get()) / 100 # 轉換為米
41
42             if weight <= 0 or height <= 0:
43                 messagebox.showerror("錯誤", "體重和身高必須為正數")
44
45             bmi = weight / (height * height)
46             self.bmi_result.config(text=f"{bmi:.1f}")
47
48             # 判斷體重狀態
49             if bmi < 18.5:
50                 status = "過輕"
51             elif bmi < 24:
52                 status = "正常"
53             elif bmi < 27:
54                 status = "過重"
55             elif bmi < 30:
56                 status = "輕度肥胖"
57             elif bmi < 35:
58                 status = "中度肥胖"
59             else:
60                 status = "重度肥胖"
61
62             self.status_result.config(text=status)
63
64         except ValueError:
65             messagebox.showerror("錯誤", "請輸入有效的數字")
66
67
68     def clear_entries(self):
69         self.weight_entry.delete(0, tk.END)
70         self.height_entry.delete(0, tk.END)
71         self.bmi_result.config(text="")
72         self.status_result.config(text="")
73
74     if __name__ == "__main__":
75         root = tk.Tk()
76         app = BMIcalculator(root)
77         root.mainloop()
78

```

總評價：

- 內容準確性：5/5
- 內容完整度：5/5
- 表達能力：5/5
- 創意性：5/5

第三位參賽者：Gemini

基礎題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請撰寫一個 Python 程式來計算費氏數列 (Fibonacci)。」

◆ 當然，以下是一個簡短的 Python 程式來計算費氏數列：

```
Python

def fibonacci(n):
    if n <= 0:
        return []
    elif n == 1:
        return [0]
    else:
        list_fib = [0, 1]
        while len(list_fib) < n:
            next_fib = list_fib[-1] + list_fib[-2]
            list_fib.append(next_fib)
        return list_fib

# 範例：計算前 10 個費氏數列
print(fibonacci(10))
```

這個程式定義了一個名為 `fibonacci` 的函式，它接受一個整數 `n` 作為輸入，並返回一個包含前 `n` 個費氏數列的列表。

```
In [ ]: def fibonacci(n):
    if n <= 0:
        return []
    elif n == 1:
        return [0]
    else:
        list_fib = [0, 1]
        while len(list_fib) < n:
            next_fib = list_fib[-1] + list_fib[-2]
            list_fib.append(next_fib)
        return list_fib

# 範例：計算前 10 個費氏數列
print(fibonacci(10))
```

[0, 1, 1, 2, 3, 5, 8, 13, 21, 34]

請用繁體中文簡短回答以下問題

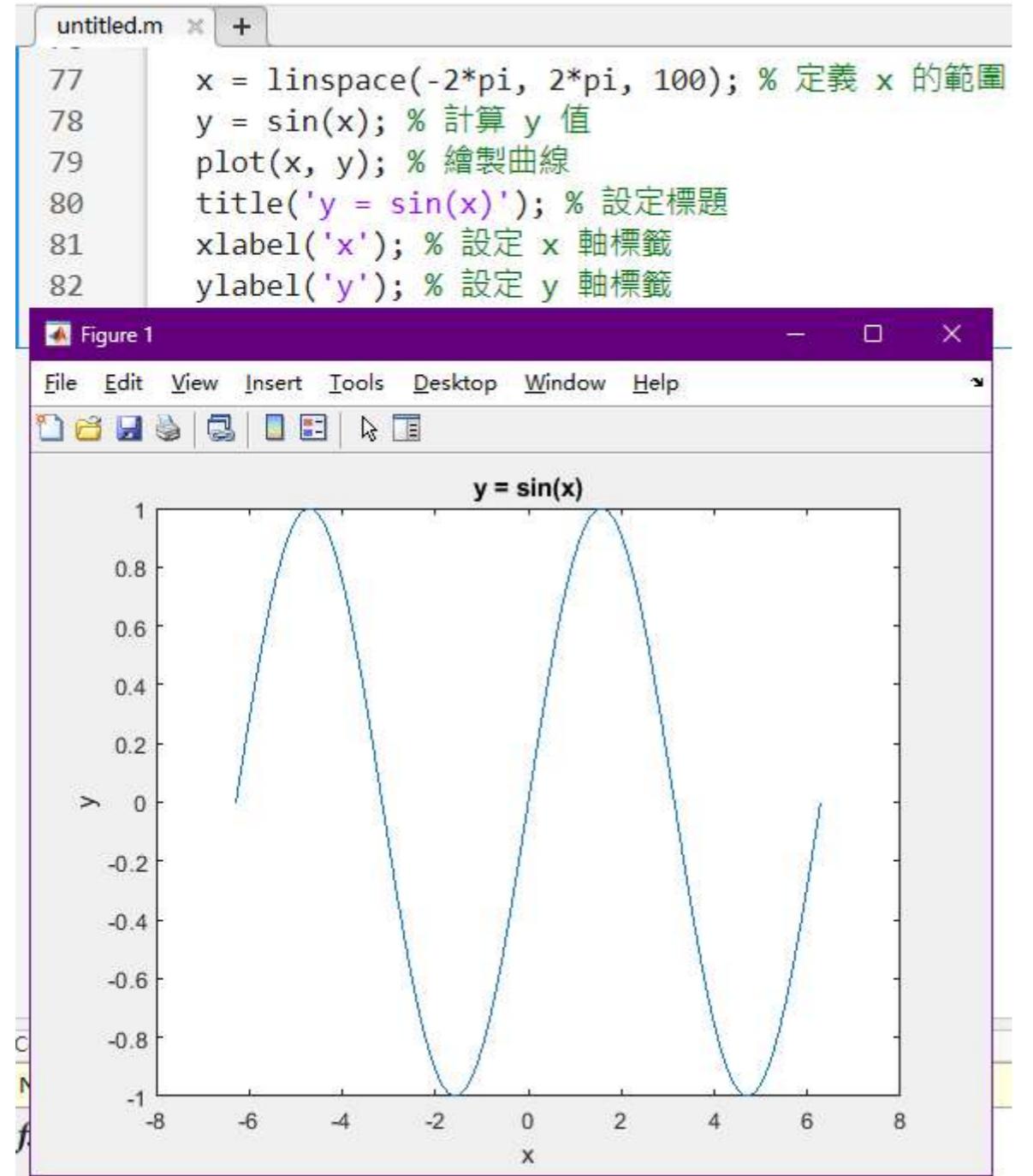
「請用 Matlab 繪製函數 $y = \sin(x)$ 的曲線。」

好的，以下是一個簡短的 Matlab 程式碼，用於繪製函數 $y = \sin(x)$ 的曲線：

Matlab

```
x = linspace(-2*pi, 2*pi, 100); % 定義 x 的範圍
y = sin(x); % 計算 y 值
plot(x, y); % 繪製曲線
title('y = sin(x)'); % 設定標題
xlabel('x'); % 設定 x 軸標籤
ylabel('y'); % 設定 y 軸標籤
```

這個程式碼會生成一個圖形視窗，顯示 $\sin(x)$ 函數在 -2π 到 2π 範圍內的曲線。



進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請用 Matlab 設計一個控制系統模擬，並分析其步階響應
(step response)。」

◆ 好的，以下是一個簡短的 Matlab 程式碼，用於設計一個控制系統模擬並分析其步階響應：

Matlab

```
% 定義系統傳遞函數
num = 1;
den = [1, 2, 1];
sys = tf(num, den);

% 繪製步階響應
step(sys);

% 設定圖表標題和軸標籤
title('控制系統步階響應');
xlabel('時間 (秒)');
ylabel('輸出');

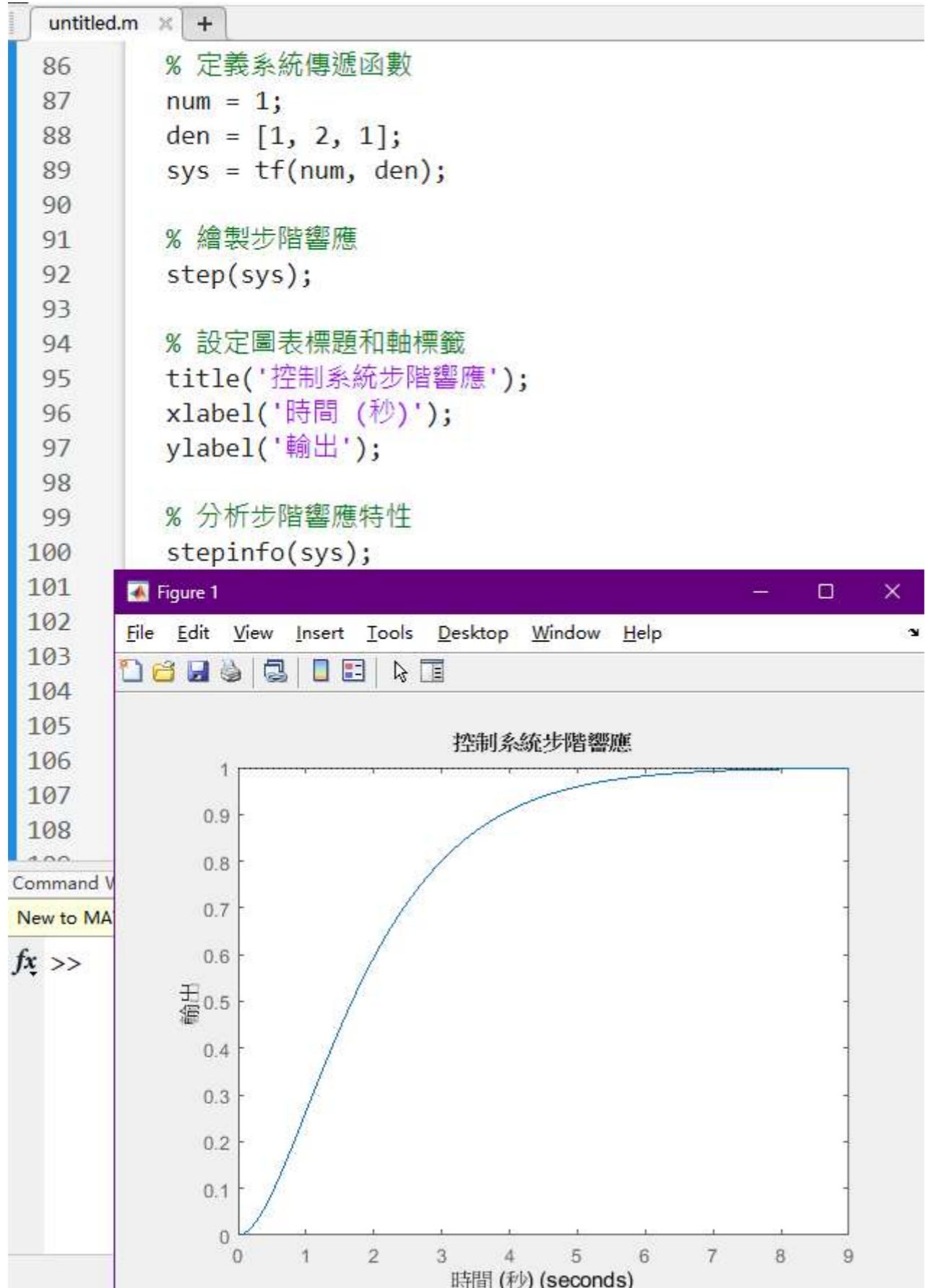
% 分析步階響應特性
stepinfo(sys);
```

這個程式碼會：

1. 定義系統傳遞函數：使用 `tf` 函數創建一個二階系統。
2. 繪製步階響應：使用 `step` 函數繪製系統對單位步階輸入的響應。
3. 設定圖表標題和軸標籤：使用 `title`、`xlabel` 和 `ylabel` 函數添加圖表標題和軸標籤。
4. 分析步階響應特性：使用 `stepinfo` 函數計算並顯示步階響應的特性，例如上升時間、穩定時間、超調量等。

執行此程式碼後，Matlab 會顯示步階響應的圖形，並在命令視窗中顯示步階響應的分析結果。

▲ □ C < :



應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

「請用 Python 設計一個簡單的 GUI 應用程式，讓使用者輸入數據並計算 BMI。」

好的，以下是一個使用 `tkinter` 模組設計的簡單 BMI 計算機 GUI 應用程式：

Python

```
import tkinter as tk

def calculate_bmi():
    height = float(height_entry.get()) / 100 # 將公分轉換為公尺
    weight = float(weight_entry.get())
    bmi = weight / (height ** 2)
    result_label.config(text=f"BMI: {bmi:.2f}")

window = tk.Tk()
window.title("BMI 計算機")

height_label = tk.Label(window, text="身高 (公分):")
height_label.pack()
height_entry = tk.Entry(window)
height_entry.pack()

weight_label = tk.Label(window, text="體重 (公斤):")
weight_label.pack()
weight_entry = tk.Entry(window)
weight_entry.pack()

calculate_button = tk.Button(window, text="計算", command=calculate_bmi)
calculate_button.pack()

result_label = tk.Label(window, text="BMI: ")
result_label.pack()

window.mainloop()
```

程式說明：

程式說明：

1. 匯入 `tkinter` 模組：導入 `tkinter` 模組以創建 GUI 介面。
2. 定義 `calculate_bmi` 函式：
 - 獲取使用者輸入的身高和體重。
 - 計算 BMI 值。
 - 更新 `result_label` 的文字以顯示 BMI 結果。
3. 創建主視窗：使用 `tk.Tk()` 創建主視窗。
4. 創建標籤和輸入框：
 - 使用 `tk.Label` 創建身高和體重標籤。
 - 使用 `tk.Entry` 創建身高和體重輸入框。
5. 創建計算按鈕：使用 `tk.Button` 創建計算按鈕，並將其 `command` 參數設定為 `calculate_bmi` 函式。
6. 創建結果標籤：使用 `tk.Label` 創建結果標籤，用於顯示 BMI 結果。
7. 運行主迴圈：使用 `window.mainloop()` 運行主迴圈，使 GUI 應用程式保持運行狀態。

執行此程式碼後，將會出現一個簡單的 BMI 計算機視窗，使用者可以在其中輸入身高和體重，然後點擊「計算」按鈕來查看 BMI 結果。



```
gemini_bmi_calculator_gui.py | D: > 台師大顏伯亨 > (2)8-10 生成式AI > GenAi_HW4 > gemini_bmi_calculator_gui.py > ...
```

```
1 import tkinter as tk
2
3     Codeium: Refactor | Explain | Generate Docstring | X
4 def calculate_bmi():
5     height = float(height_entry.get()) / 100 # 將公分轉換為公尺
6     weight = float(weight_entry.get())
7     bmi = weight / (height ** 2)
8     result_label.config(text=f"BMI: {bmi:.2f}")
9
10 window = tk.Tk()
11 window.title("BMI 計算機")
12 height_label = tk.Label(window, text="身高 (公分):")
13 height_label.pack()
14 height_entry = tk.Entry(window)
15 height_entry.pack()
16
17 weight_label = tk.Label(window, text="體重 (公斤):")
18 weight_label.pack()
19 weight_entry = tk.Entry(window)
20 weight_entry.pack()
21
22 calculate_button = tk.Button(window, text="計算", command=calculate_bmi)
23 calculate_button.pack()
24
25 result_label = tk.Label(window, text="BMI: ")
26 result_label.pack()
27
28 window.mainloop()
```



總評價：

- 內容準確性：5/5
- 內容完整度：4/5
- 表達能力：5/5
- 創意性：4/5

測試關卡(四)：程式除錯 (C# 、 Python 、 Matlab)

基礎題：

- 這段 C# 程式碼有錯誤，請修正：

```
int x = null;
Console.WriteLine(x);
```

- 這段 Python 程式碼有錯誤，請找出並修正：

```
num = "10"
result = num + 5
print(result)
```

進階題：

- 這段 Python 程式碼會造成記憶體洩漏 (memory leak)，請找出問題並修正：

```
lst = []
while True:
    lst.append("leak")
```

應用題：

- 這段 Matlab 程式碼運行效率低，請優化：

```
A = rand(20000, 20000);
B = zeros(20000, 1);
for i = 1:20000
    B(i) = sum(A(i, :));
end
```

第一位參賽者：ChatGPT

基礎題回答內容截圖，以及輸出內容截圖



Program.cs

```
1  using System;
2  using System.Threading;
3  using System.Threading.Tasks;
4
5  class Program
6  {
7      static void Main()
8      {
9          int? x = null;
10         Console.WriteLine(x);
11     }
12 }
13
14
15
16
17
18
19
20
21
```

找不到任何問題

Microsoft Visual Studio

D:\台師大顏伯亨\2023-10 生成式AI\GenAi_HW4\Csharp\ConsoleApp1\bin\Debug\ConsoleApp1.exe (流程 10524) 已結束，代碼為 0 (0x0)。

若要在偵錯停止時自動關閉主控台，請啟用 [工具] -> [選項] -> [偵錯] -> [偵錯停止時，自動關閉主控台]。

按任意鍵關閉此視窗...

顯示輸出來源(S): 偵錯

ConsoleApp1.exe' (CLR v4.0.30319: DefaultDomain): 已載入 'C:\WINDOWS\Microsoft.NET\assembly\GAC_32\mscorlib\w4.0_4.0.0.0_b77a5c561934e089\mscorlib.dll'。已略過載入符號。模組已最佳化，並已啟用 [Just In] 偵錯。
ConsoleApp1.exe' (CLR v4.0.30319: DefaultDomain): 已載入 'D:\台師大顏伯亨\2023-10 生成式AI\GenAi_HW4\Csharp\ConsoleApp1\bin\Debug\ConsoleApp1.exe'。已載入符號。
[10524] ConsoleApp1.exe' 程式以返回碼 0 (0x0) 結束。



```
In [ ]: num = "10"  
        result = int(num) + 5  
        print(result)
```

15

進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

這段 Python 程式碼會造成記憶體洩漏 (memory leak)，請找出問題並修正：

```
lst = []
while True:
    lst.append("leak")
```

問題：無限迴圈持續向 `lst` 添加元素，導致記憶體洩漏。

修正方法：添加適當的退出條件，或使用生成器 (`yield`) 來避免持續佔用記憶體。

修正範例 1 (加退出條件)：

```
python
lst = []
for _ in range(1000): # 限制迴圈次數
    lst.append("leak")
```

修正範例 2 (使用生成器)：

```
python
def leak_generator():
    while True:
        yield "leak"

gen = leak_generator()
for _ in range(1000): # 控制迴圈次數
    next(gen)
```

In []: #原本的錯誤程式碼，會導致記憶體用盡的錯誤

```
lst = []
while True:
    lst.append("leak")
```

```
KeyboardInterrupt                                     Traceback (most recent call last)
<ipython-input-37-d3d8dad87ff3> in <cell line: 0>()
      2
      3 lst = []
----> 4 while True:
      5     lst.append("leak")
```

KeyboardInterrupt:

```
In [ ]: #修正範例1
lst = []
for _ in range(1000): # 限制迴圈次數
    lst.append("leak")
```

```
In [ ]: #修正範例2

def leak_generator():
    while True:
        yield "leak"

gen = leak_generator()
for _ in range(1000): # 控制迴圈次數
    next(gen)
```

應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

這段 Matlab 程式碼運行效率低，請優化：

```
A = rand(20000, 20000);
B = zeros(20000, 1);
for i = 1:20000
    B(i) = sum(A(i,:));
end
```

問題：`for` 迴圈逐行累加矩陣 `A`，運行效率低。

優化方法：利用 MATLAB 向量化運算，提高效率。

優化後程式碼：



```
matlab
A = rand(20000, 20000);
B = sum(A, 2); % 直接對矩陣 A 按列求和
```

這樣可大幅提升計算速度，避免 `for` 迴圈的額外開銷。

The screenshot shows the MATLAB environment. The Editor window at the top contains the script file `untitled2.m` with the following code:

```
1  tic; % 開始計時
2
3  A = rand(20000, 20000);
4  B = zeros(20000, 1);
5  for i = 1:20000
6      B(i) = sum(A(i, :));
7  end
8
9  toc; % 顯示執行時間
10 %% chatgpt
11
12 tic;
13 A = rand(20000, 20000);
14 B = sum(A, 2); % 直接對矩陣 A 按列求和
15 toc;
16
17 %% claude
18
19
20 %% gemini
```

The Command Window below shows the output of the script:

```
Command Window
New to MATLAB? See resources for Getting Started.
Elapsed time is 8.842006 seconds.
fx >>
```

上圖為執行原本的程式碼，花的時間為8.842秒

The screenshot shows the MATLAB Editor and Command Window. The Editor window displays a script named 'untitled2.m' with the following content:

```
1 tic; % 開始計時
2
3 A = rand(20000, 20000);
4 B = zeros(20000, 1);
5 for i = 1:20000
6     B(i) = sum(A(i, :));
7 end
8
9 toc; % 顯示執行時間
10 %% chatgpt
11
12 tic;
13 A = rand(20000, 20000);
14 B = sum(A, 2); % 直接對矩陣 A 按列求和
15 toc;
16
17 %% claude
18
19
20 %% gemini
```

The Command Window below shows the execution results:

```
Command Window
New to MATLAB? See resources for Getting Started.
Elapsed time is 8.842006 seconds.
Elapsed time is 3.255079 seconds.
```

上圖為執行改善的程式碼，花的時間為3.255秒

總評價：

- 內容準確性：5/5
- 內容完整性：3/5
- 表達能力：5/5
- 創意性：4/5

第二位參賽者：Claude

基礎題回答內容截圖，以及輸出內容截圖

The screenshot shows a user interface for a programming challenge. At the top, there is a message in Chinese: "請用繁體中文簡短回答以下問題" (Please answer the following question in simplified Chinese) and a note: "* 這段 C# 程式碼有錯誤，請修正：" (This C# code has an error, please fix it). Below this is a code editor window with the following code:

```
csharp
int x = null;
Console.WriteLine(x);
```

Next, there is a detailed explanation in Chinese: "這段 C# 程式碼的問題在於試圖將 null 值賦給 int 型別，而 int 是值類型，不能接受 null 值。若要允許 int 型別接受 null 值，需要使用可空類型 (nullable type) 。" (The problem with this C# code is that it tries to assign a null value to an int type, while int is a value type and cannot accept null values. To allow int to accept null values, you need to use a nullable type.)

Below the explanation, there is a note: "修正後的程式碼如下：" (The corrected code is as follows:)

```
csharp
int? x = null;
Console.WriteLine(x);
```

Then, there is another note: "或者，如果你想要一個有預設值的整數：" (Or, if you want an integer with a default value:)

```
csharp
int x = 0; // 或其他預設值
Console.WriteLine(x);
```

At the bottom, there is a message from Claude: "Claude does not have the ability to run the code it generates yet." (Claude does not have the ability to run the code it generates yet.)

(第一個修正程式碼和ChatGPT相同，所以就只有附上第二個修正程式碼)

The screenshot shows the Microsoft Visual Studio IDE interface. The top navigation bar displays 'Program.cs' and 'ConsoleApp1'. The code editor window contains the following C# code:

```
1  using System;
2  using System.Threading;
3  using System.Threading.Tasks;
4
5  class Program
6  {
7      static void Main()
8      {
9          int x = 0; // 或其他預設值
10         Console.WriteLine(x);
11     }
12 }
13
14
15
16
17
18
19
20
21
```

The line 'int x = 0;' is highlighted with a red rectangle. A tooltip window titled 'Microsoft Visual Studio' is displayed over this line, showing the value '0'.

The status bar at the bottom left indicates '101 %' and '找不到任何問題' (No problems found). The output window at the bottom shows the build log:

```
顯示輸出來源(S): 建置
組建已在下午 09:57 開始...
1>----- 已開始建置: 專案: ConsoleApp1, 設定: Debug Any CPU -----
1> ConsoleApp1 -> D:\台師大顏伯亨\2023-10 生成式AI\GenAi_HW4\Csharp\ConsoleApp1\bin\Debug\ConsoleApp1.exe
===== 組建: 1 成功, 0 失敗, 0 為最新狀態, 0 已跳過 =====
===== 組建 已於下午 09:57 完成並花了 01.052 秒 =====
```

嚴 請用繁體中文簡短回答以下問題

- 這段 Python 程式碼有錯誤，請找出並修正：

`num = "10"
result = num + 5
print(result)`

這段 Python 程式碼的錯誤在於試圖將字串 ("10") 和數字 (5) 相加，這會導致型別錯誤。在 Python 中，不能直接將字串和數字相加。

修正後的程式碼如下：

`python
num = "10"
result = int(num) + 5
print(result)`

修正方法是將字串 "10" 轉換為整數型別，使用 `int()` 函數進行轉換，然後再與數字 5 相加。
這樣執行結果會輸出 15。

① Claude does not have the ability to run the code it generates yet.

Claude can make mistakes. Please double-check responses.



```
In [ ]: num = "10"  
result = int(num) + 5  
print(result)
```

進階題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

這段 Python 程式碼會造成記憶體洩漏 (memory leak)，請找出問題並修正：

```
lst = []
while True:
    lst.append("leak")
```

這段 Python 程式碼會造成記憶體洩漏，因為它在一個無限迴圈中不斷地向清單 `lst` 中添加字串 "leak"，而沒有任何停止條件或清理機制。

問題所在：

1. 無限迴圈 `while True` 沒有退出條件
2. 清單 `lst` 會無限擴大，最終耗盡記憶體

修正後的程式碼可以是：

`python`

```
lst = []
max_size = 1000 # 設定一個最大大小
while len(lst) < max_size:
    lst.append("leak")
```

或者，如果需要在一個長時間運行的程序中處理這種情況：

`python`

```
lst = []
while True:
    lst.append("leak")
    if len(lst) >= 1000:
        lst = [] # 清空列表，避免記憶體洩漏
```

這兩種方法都能防止記憶體洩漏，一個是設定停止條件，另一個是定期清空列表。

① Claude does not have the ability to run the code it generates yet.

In []: #Claude 修正的程式碼-1

```
lst = []
max_size = 1000 # 設定一個最大大小
while len(lst) < max_size:
    lst.append("leak")
```

In []: #Claude 修正的程式碼-2

```
lst = []
while True:
    lst.append("leak")
```

```
if len(lst) >= 1000:  
    lst = [] # 清空列表，避免記憶體洩漏
```

應用題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題
這段 Matlab 程式碼運行效率低，請優化：

```
A = rand(20000, 20000);  
B = zeros(20000, 1);  
for i = 1:20000  
    B(i) = sum(A(i, :));  
end
```

這段 Matlab 程式碼效率低的原因是使用了迴圈來處理矩陣操作，而 Matlab 在向量化操作上效能更好。

優化後的程式碼：

```
matlab  
A = rand(20000, 20000);  
B = sum(A, 2);
```

這個優化使用了 Matlab 的內建函數 `sum` 並指定第二個參數為 2，表示沿著每行 (row) 進行求和。這種向量化操作比迴圈遍歷每一行要快得多，因為：

1. 避免了迴圈的開銷
2. 利用了 Matlab 的矩陣運算優化
3. 可能使用了並行計算

這樣的寫法不僅程式碼更簡潔，而且效率會顯著提高，特別是在處理大型矩陣時。

(① Claude does not have the ability to run the code it generates yet.)

Claude can make mistakes. Please double-check responses.

(和ChatGPT修正的程式碼相同)

總評價：

- 內容準確性：5/5
- 內容完整度：5/5
- 表達能力：5/5
- 創意性：5/5

第三位參賽者：Gemini

基礎題回答內容截圖，以及輸出內容截圖

請用繁體中文簡短回答以下問題

這段 C# 程式碼有錯誤，請修正：

```
int x = null;
Console.WriteLine(x);
```

這段程式碼的錯誤在於：

- `int x = null;` : `int` 是一個數值型別，不能直接賦予 `null` 值。如果需要允許 `null` 值，可以使用可為 `null` 的數值型別 `int?`。

修正後的程式碼如下：

C#

```
int? x = null;
Console.WriteLine(x);
```

△ □ C < :

請用繁體中文簡短回答以下問題

這段 Python 程式碼有錯誤，請找出並修正：

```
num = "10"
result = num + 5
print(result)
```

◆ 這段程式碼的錯誤在於：

- `num + 5` : `num` 是一個字串，而 `5` 是一個整數，在 Python 中不能直接將字串和整數相加。

修正方法：

1. 將 `num` 轉換為整數：

Python

```
num = "10"
result = int(num) + 5
print(result)
```

2. 將 5 轉換為字串

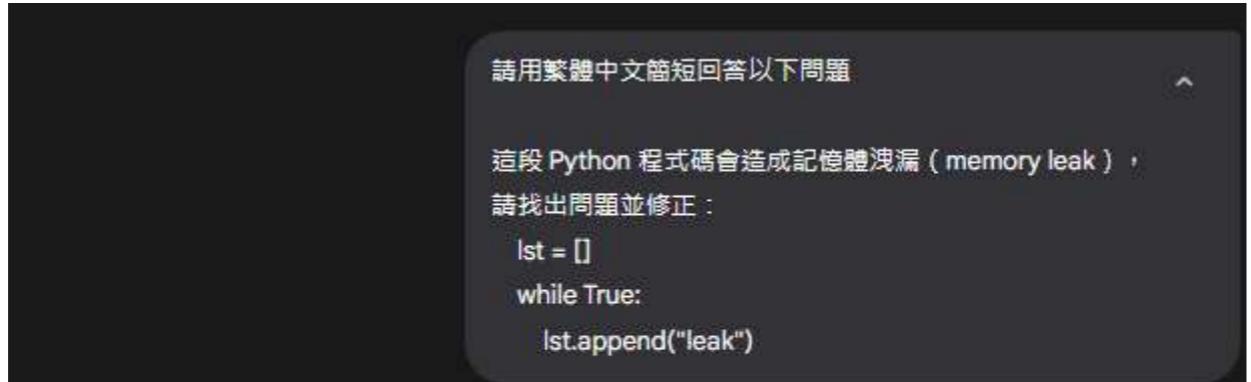
Python

```
num = "10"
result = num + str(5)
print(result)
```

```
In [ ]: num = "10"
result = int(num) + 5
print(result)
```

```
In [ ]: num = "10"
result = num + str(5)
print(result)
```

進階題回答內容截圖，以及輸出內容截圖



◆ 這段程式碼的問題在於 `while True` 無限迴圈中不斷將 "leak" 字串添加到 `lst` 清單中，導致清單無限增長，最終耗盡記憶體。

修正方法：

1. 限制迴圈次數：設定一個條件來終止迴圈。

```
Python

lst = []
count = 0
while count < 1000: # 例如：限制迴圈 1000 次
    lst.append("leak")
    count += 1
```

2. 設定清單最大長度：如果需要無限迴圈，可以設定清單的最大長度，並在達到最大長度時刪除舊元素。

```
Python

lst = []
max_length = 1000 # 設定最大長度
while True:
    lst.append("leak")
    if len(lst) > max_length:
        del lst[0] # 刪除第一個元素
```

In []: #第一個修正方案：限制迴圈次數

```
lst = []
count = 0
while count < 1000: # 例如：限制迴圈 1000 次
    lst.append("leak")
    count += 1
```

In []: #第二個修正方案：設定list的最大長度

```
lst = []
```

```
max_length = 1000 # 設定最大長度
while True:
    lst.append("leak")
    if len(lst) > max_length:
        del lst[0] # 刪除第一個元素
```

應用題回答內容截圖，以及輸出內容截圖



(和ChatGPT與Claude修正的程式碼相同)

總評價：

- 內容準確性：5/5
- 內容完整度：4/5
- 表達能力：4/5
- 創意性：4/5

====評估模型表現=====

使用ChatGPT依照評分結果畫出長條圖

可以依照以下的評分劃出直條圖嗎，橫軸為LLM名稱，分四種顏色，"accuracy", "completeness", "Expressive ability", "Creativity"，並且依照測試分四張圖：

測試1. 自動控制系統觀念

1. ChatGPT (GPT-4o mini)

* 內容準確性 : 5/5

* 內容完整性 : 4/5

* 表達能力 : 5/5

* 創意性 : 4/5

2. Claude (Claude 3.7 Sonnet)

內容準確性 : 5/5

內容完整性 : 5/5

表達能力 : 5/5

創意性 : 5/5

3. Gemini (Google Gemini 2.0 Flash)

內容準確性 : 5/5

內容完整性 : 3/5

表達能力 : 4/5

創意性 : 3/5

測試2. 材料力學概念

1. ChatGPT (GPT-4o mini)

內容準確性 : 5/5

內容完整性 : 4/5

表達能力 : 4/5

創意性 : 4/5

2. Claude (Claude 3.7 Sonnet)

內容準確性 : 5/5

內容完整性 : 5/5

表達能力 : 4/5

創意性 : 5/5

3. Gemini (Google Gemini 2.0 Flash)

內容準確性 : 4/5

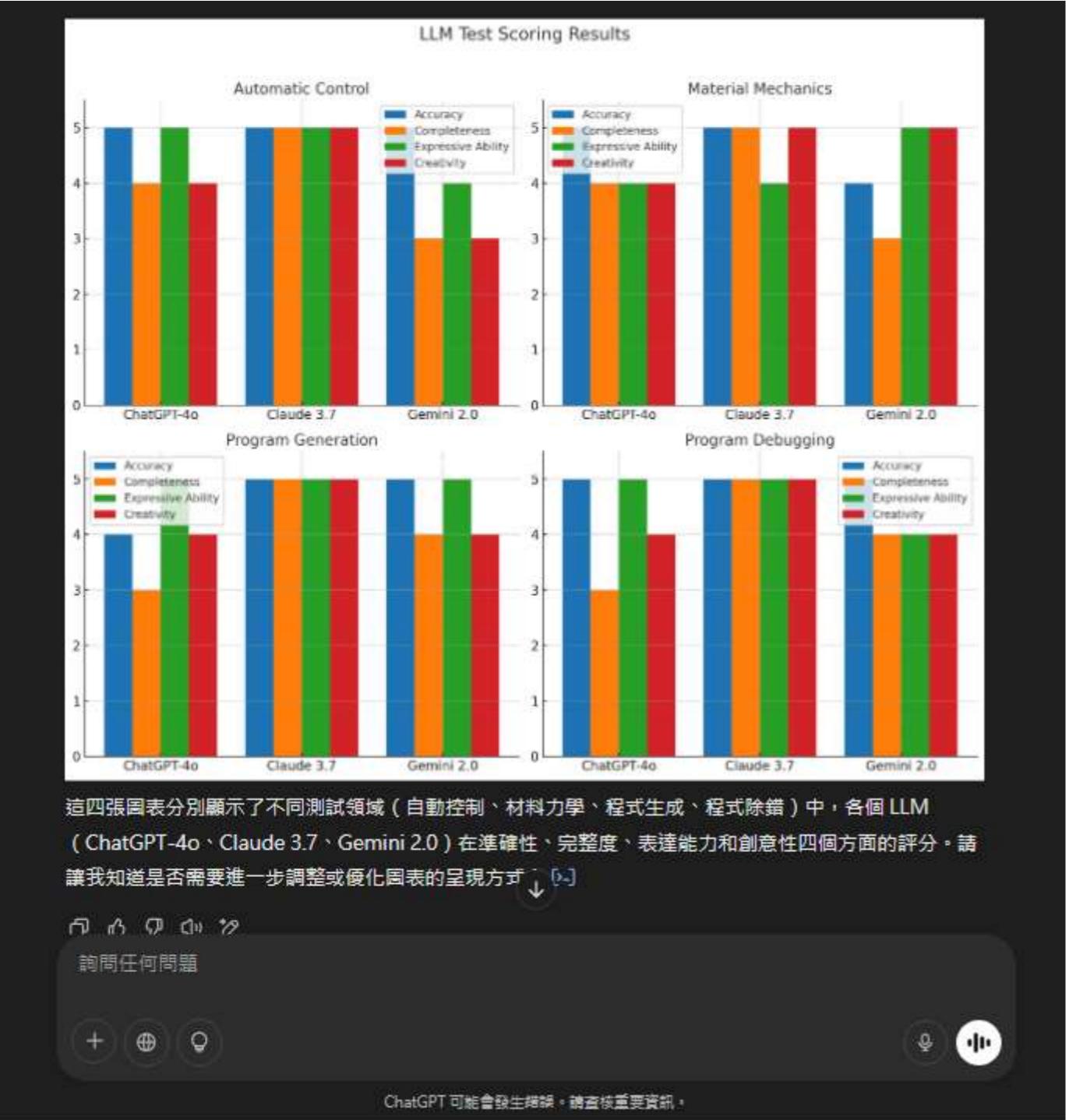
內容完整性 : 3/5

表達能力 : 5/5

創意性 : 5/5



測試3. 程式生成(C#,Python,Matlab)



```
In [28]: plt.rcParams['font.family'] = 'DejaVu Sans'

categories = ["Accuracy", "Completeness", "Expressive Ability", "Creativity"]
models = ["ChatGPT-4o", "Claude 3.7", "Gemini 2.0"]
tests = ["Automatic Control", "Material Mechanics", "Program Generation", "Program Debugging"]

scores = {
    "ChatGPT-4o": [
        [5, 4, 5, 4], # 自動控制系統觀念
        [5, 4, 4, 4], # 材料力學概念
        [4, 3, 5, 4], # 程式生成
        [5, 3, 5, 4], # 程式除錯
    ],
    "Claude 3.7": [
        [5, 5, 5, 5],
        [5, 5, 4, 5],
        [5, 5, 5, 5],
        [5, 5, 5, 5]
    ],
    "Gemini 2.0": [
        [4, 3, 4, 4],
        [3, 3, 4, 4],
        [4, 4, 5, 4],
        [4, 4, 5, 4]
    ]
}
```

```
[ 5,  5,  5,  5],
[ 5,  5,  5,  5],
],
"Gemini 2.0": [
[ 5,  3,  4,  3],
[ 4,  3,  5,  5],
[ 5,  4,  5,  4],
[ 5,  4,  4,  4],
],
},
}

colors = ["#1f77b4", "#ff7f0e", "#2ca02c", "#d62728"]

fig, axes = plt.subplots(2, 2, figsize=(12, 10))
fig.suptitle("LLM Test Scoring Results", fontsize=16)

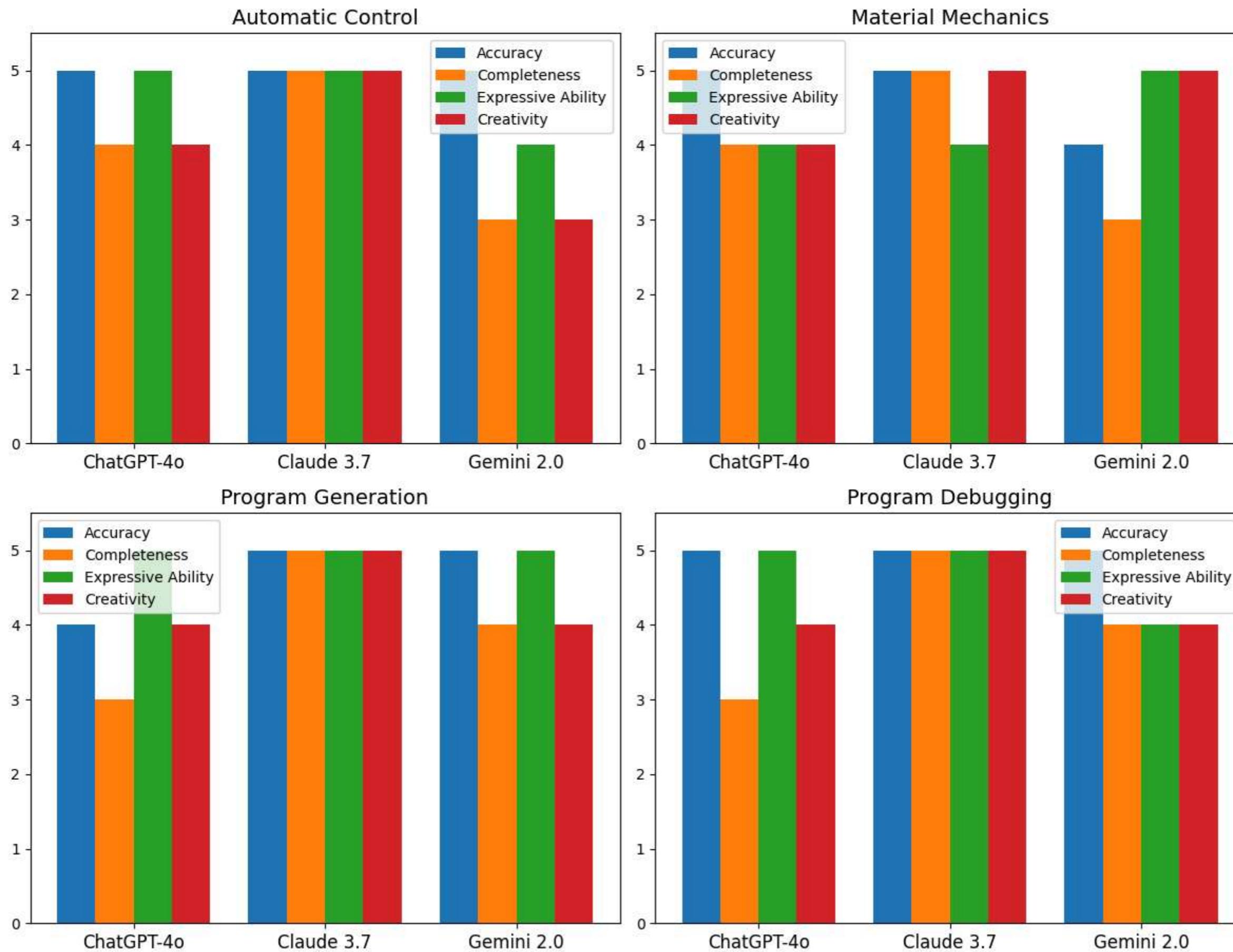
for i, test in enumerate(tests):
    ax = axes[i // 2, i % 2]
    x = np.arange(len(models))
    width = 0.2

    for j, category in enumerate(categories):
        scores_for_category = [scores[model][i][j] for model in models]
        ax.bar(x + j * width - 0.3, scores_for_category, width, label=category, color=colors[j])

    ax.set_title(test, fontsize=14)
    ax.set_xticks(x)
    ax.set_xticklabels(models, fontsize=12)
    ax.set_xlim(0, 5.5)
    ax.legend()

plt.tight_layout(rect=[0, 0, 1, 0.96])
plt.show()
```

LLM Test Scoring Results



透過圖表分析每個LLM的優劣勢

LLM	最佳領域	優勢	缺點
Claude 3.7	全能型 AI，擅長所有領域	高準確度、完整度強、表達能力優秀、創意高	可能過於詳盡，回應過長
ChatGPT-4o	適合需要清楚解釋的程式與工程概念	高準確度、表達能力好	完整度稍低，創意不及 Claude
Gemini 2.0	程式生成與材料力學表達能力佳	表達能力與創意良好，程式相關能力穩定	完整度較低，部分領域準確性稍低

總結

- Claude 3.7：需要最完整、最精確的答案（適合技術研究、學術應用）
- ChatGPT-4o：獲得清楚的解釋，且不希望回答過長（適合教學與學習）
- Gemini 2.0：關注程式開發，並希望獲得不同的創意思路

喜歡哪一個？

經過一段時間的使用及測試，我個人最喜歡 Claude，特別是在寫程式時。Claude 的強大之處在於，它不僅能夠準確地生成程式碼，還經常會額外提供一些意想不到的實用功能，讓開發變得更加高效。而且，它的錯誤率極低，這讓我在編寫與測試程式時更加放心。